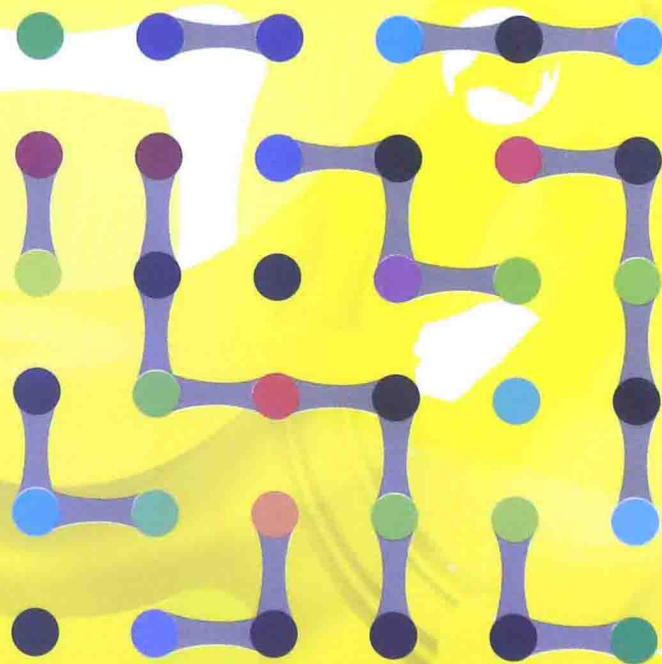


# 从0开始教你学 STM32

赵星寒 编著



北京航空航天大学出版社  
BEIHANG UNIVERSITY PRESS

# 从 0 开始教你学 STM32

赵星寒 编著

北京航空航天大学出版社

## 内 容 简 介

本书以引导读者快速全面掌握 STM32 系列嵌入式微控制器为目的,由浅入深地带领大家进入 STM32 的世界。本书详细介绍了 STM32F103 最小系统、寄存器及存储器、最小系统管理、中断和事件、汇编语言编程方法、汇编语言工程项目、Thumb-2 指令集、伪指令、C 语言编程方法、混合编程方法、通用外部端口 GPIO 和复用外部端口 AFIO、模拟/数字转换器 ADC、通用异步同步收发器 USART 等。

本书非常适合 STM32F10x 嵌入式微控制器的初学者,以及有一定嵌入式应用基础的电子工程技术人员参考,也可作为高等院校电子信息、自动控制等专业的教学和科研开发参考书。

### 图书在版编目(CIP)数据

从 0 开始教你学 STM32 / 赵星寒编著. -- 北京:北京航空航天大学出版社,2014.9

ISBN 978-7-5124-1567-6

I. ①从… II. ①赵… III. ①微控制器 IV.

①TP332.3

中国版本图书馆 CIP 数据核字(2014)第 170812 号

版权所有,侵权必究。



### 从 0 开始教你学 STM32

赵星寒 编著

责任编辑 潘晓丽 刘秀清 张雯佳

\*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316524

北京楠海印刷厂印装 各地书店经销

\*

开本:710×1 000 1/16 印张:18 字数:384 千字

2014 年 9 月第 1 版 2014 年 9 月第 1 次印刷 印数:3 000 册

ISBN 978-7-5124-1567-6 定价:39.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

# 前 言

---

我学过和使用过很多种微处理器,但还是对 STM32 系列情有独钟,在应用了几次这种芯片之后,很想向广大初学者推荐这个系列产品。

STM32 系列是一款以 ARM 为核的微处理器系列,其内核是高档的 ARM Cortex-M3 体系结构。STM32 系列微处理器成员比较多,既包含高性能外围复杂的芯片,也包含高性能小封装的芯片,其中小封装的芯片以价格低廉、性能优良而著称,在很多场合可以替代高端的单片机。

学习这个系列的微处理器,对于初学者建议如下。

## 1. 使用寄存器编程方法,而不是函数库编程方法

STM32 系列软件编程有两种方法:一种是传统的编程方法,称为寄存器方法,就是学习使用微处理器的寄存器,并通过对寄存器的操作实现编程;另一种称为函数库方法,就是使用厂家提供的函数库编程,这种方法不需要学习微处理器的寄存器。

之所以推荐使用寄存器方法,主要原因如下:

① 寄存器编程方法是普遍应用的方法。这种编程方法,对所有微处理器都适用,因此学会这种方法对学习其他微处理器会有所帮助;而函数库编程方法只是针对这一系列微处理器适用,对学习其他微处理器则没有帮助。

② 寄存器编程方法是一种底层的、与硬件相关的方法。在使用中需要对微处理器的硬件和寄存器进行认真的学习,有助于了解各种微处理器及其应用;而即使能够很熟练地应用函数库方法,也无法了解微处理器的细节,对深入学习没有多大帮助。

③ 如果对微处理器的寄存器了解不够,那么在使用函数库方法时遇到困难就无法解决。

总之,寄存器编程方法是最基本的方法,具有普遍性,因此本书建议初学者从学习这一方法出发,以 STM32 微处理器为入门芯片,开始学习微处理器的应用,特别是 ARM 系列微处理器的应用。

## 2. 学习微处理器,重要的是实践,因此需要找一套仿真环境软件

仿真环境建议使用 Keil 公司的  $\mu$ Vision,或者称为 RealView,这套软件易于学

## 前言

习,一般试用版也较易获得。在这个环境中,可以编辑、编译 C 语言和汇编语言源程序,也可以仿真调试程序,提高自己的实战水平。

在使用时,注意需要正确设置环境。

### 3. 如果熟悉了上述仿真软件,还需要找一套仿真器

现在,一套 STM32 的仿真器价格很低,这是学习的基础。一般情况下,STM32 芯片的仿真器可以仿真很多芯片,因此还会有很多其他用途。仿真器都是 JTAG 接口的,结构简单。

### 4. 实验产品或实验板

自己可以提出一个目标产品作为学习实验使用,目标产品不要太复杂,从原理设计到 PCB 板制作,从焊接到调试,完成一个完整的研发过程,对学习是十分有帮助的;也可以自己购买一块实验板,这样成本会低一些,购买实验板时,建议只购买最简单的实验板,这样的实验板上只有一片 STM32 芯片,不需要其他外部附件。

在这个实验板的基础上,飞线连接一些基本部件,比如键盘、LED 管,使用电位器为 A/D 实验提供信号源、外部开关等,非常简单而实用。

如果有问题需要讨论,请按照下列地址与作者联系:zxhjeket@aliyun.com。

感谢我的同事郑玉峰、周春来、刘晓波、赵丽、孙蓬蛟、王庚兰、赵泽以及我的朋友吴乌恩、程宁等的支持。

赵星寒

2014 年 7 月于北京



|                              |    |
|------------------------------|----|
| 第 1 章 概 述 .....              | 1  |
| 1.1 STM32F1xx 系列概述 .....     | 1  |
| 1.1.1 STM32F1xx 的多个子系列 ..... | 1  |
| 1.1.2 STM32F1xx 系列特点 .....   | 2  |
| 1.1.3 内核简介 .....             | 3  |
| 1.2 封 装 .....                | 3  |
| 1.3 接 口 .....                | 4  |
| 1.4 选型指南 .....               | 5  |
| 1.5 开发环境 .....               | 7  |
| 第 2 章 STM32F103 最小系统 .....   | 9  |
| 2.1 最小系统设计 .....             | 9  |
| 2.1.1 最小系统 .....             | 9  |
| 2.1.2 电 源 .....              | 10 |
| 2.1.3 复位控制 .....             | 11 |
| 2.1.4 系统启动模式 .....           | 12 |
| 2.1.5 振荡器 .....              | 12 |
| 2.2 JTAG 接口 .....            | 13 |
| 第 3 章 寄存器及存储器 .....          | 14 |
| 3.1 内部通用工作寄存器 .....          | 14 |
| 3.2 STM32F1xx 存储器空间 .....    | 19 |

## 目 录

|                   |                   |           |
|-------------------|-------------------|-----------|
| 3.2.1             | 存储器映像             | 19        |
| 3.2.2             | 外设管理寄存器           | 20        |
| 3.2.3             | 内嵌 Flash 存储器      | 22        |
| 3.2.4             | 内嵌 SRAM 存储器       | 23        |
| 3.2.5             | 系统存储器             | 23        |
| 3.3               | 堆栈及栈区             | 24        |
| 3.4               | 数据格式              | 25        |
| 3.4.1             | 数据的大端和小端          | 25        |
| 3.4.2             | 地址对准              | 26        |
| 3.5               | 位段(bit-banding)操作 | 28        |
| 3.5.1             | 位操作概念和位操作分区       | 28        |
| 3.5.2             | 位段操作              | 30        |
| 3.5.3             | 位段操作应用举例          | 32        |
| <b>第4章 最小系统管理</b> |                   | <b>36</b> |
| 4.1               | 电源系统管理            | 36        |
| 4.1.1             | 电压监测器 PVD         | 37        |
| 4.1.2             | 低功耗模式             | 38        |
| 4.1.3             | 实时时钟的自动唤醒 AWU 方式  | 40        |
| 4.1.4             | 电源控制寄存器组介绍        | 41        |
| 4.2               | 复位系统              | 42        |
| 4.2.1             | 系统复位              | 43        |
| 4.2.2             | 系统复位源的识别和应用       | 44        |
| 4.2.3             | 外部设备复位            | 46        |
| 4.3               | 时钟系统管理            | 49        |
| 4.3.1             | 独立看门狗(IWDG)时钟     | 50        |
| 4.3.2             | 实时时钟 RTC          | 51        |
| 4.3.3             | 系统主时钟             | 53        |
| 4.3.4             | 外部设备时钟            | 59        |
| 4.3.5             | 输出时钟源             | 62        |
| 4.3.6             | 时钟管理寄存器组          | 63        |

|                                    |     |
|------------------------------------|-----|
| 第 5 章 中断和事件 .....                  | 66  |
| 5.1 中断和事件概述 .....                  | 66  |
| 5.2 中断向量表 .....                    | 68  |
| 5.3 内核中断 .....                     | 70  |
| 5.3.1 内核中断介绍 .....                 | 71  |
| 5.3.2 内核中断的应用 .....                | 71  |
| 5.3.3 内核中断的管理 .....                | 72  |
| 5.4 外部中断/事件 .....                  | 75  |
| 5.4.1 外部中断/事件概述 .....              | 76  |
| 5.4.2 外部中断/事件管理 .....              | 78  |
| 5.5 外设中断 .....                     | 81  |
| 5.5.1 外设中断概述 .....                 | 82  |
| 5.5.2 系统中断设置/使能寄存器 .....           | 82  |
| 5.6 中断优先权和中断过程 .....               | 84  |
| 5.6.1 中断优先权寄存器 .....               | 85  |
| 5.6.2 中断过程 .....                   | 86  |
| 第 6 章 汇编语言编程方法 .....               | 89  |
| 6.1 汇编语言程序组成 .....                 | 90  |
| 6.1.1 汇编语言程序 .....                 | 90  |
| 6.1.2 汇编语言程序要素 .....               | 92  |
| 6.2 汇编语言书写规则 .....                 | 97  |
| 6.3 汇编语言中的表达式 .....                | 97  |
| 6.3.1 变 量 .....                    | 97  |
| 6.3.2 字符串表达式 .....                 | 98  |
| 6.3.3 数字表达式 .....                  | 100 |
| 6.3.4 逻辑表达式 .....                  | 102 |
| 6.3.5 以 PC 为基址和以寄存器为基址的地址表达式 ..... | 104 |
| 6.3.6 编译时的运算顺序 .....               | 105 |
| 6.4 第 2 操作数 .....                  | 105 |
| 6.4.1 第 2 操作数表达 .....              | 105 |
| 6.4.2 第 2 操作数规范 .....              | 107 |

## 目 录

|            |                          |            |
|------------|--------------------------|------------|
| 6.5        | 地址的前缀和后缀 .....           | 107        |
| 6.6        | 指令可选后缀 .....             | 108        |
| 6.7        | 条件指令的应用(条件后缀) .....      | 109        |
| 6.7.1      | 条件标志的变化 .....            | 110        |
| 6.7.2      | 条件标志位 .....              | 110        |
| 6.7.3      | 条件的判断和程序的转移(条件后缀) .....  | 111        |
| 6.8        | 数据栈的应用 .....             | 113        |
| <b>第7章</b> | <b>汇编语言工程项目</b> .....    | <b>115</b> |
| 7.1        | 汇编语言启动代码 .....           | 115        |
| 7.1.1      | 标准的启动代码 .....            | 116        |
| 7.1.2      | 自编的启动代码 .....            | 125        |
| 7.2        | 汇编语言工程项目 .....           | 126        |
| 7.2.1      | 建立一个工程项目 .....           | 126        |
| 7.2.2      | 编制主程序 .....              | 129        |
| 7.2.3      | 建立一个中断处理文件 .....         | 133        |
| 7.2.4      | 建立一个子文件 .....            | 136        |
| <b>第8章</b> | <b>Thumb-2 指令集</b> ..... | <b>138</b> |
| 8.1        | 存储器访问指令 .....            | 139        |
| 8.2        | 通用数据处理指令 .....           | 147        |
| 8.3        | 乘法和除法指令 .....            | 155        |
| 8.4        | 饱和运算指令 SSAT 和 USAT ..... | 157        |
| 8.5        | 位段操作指令 .....             | 160        |
| 8.6        | 分支和分支控制指令 .....          | 163        |
| 8.7        | 其他指令 .....               | 167        |
| <b>第9章</b> | <b>伪指令</b> .....         | <b>171</b> |
| 9.1        | 概 述 .....                | 171        |
| 9.2        | 符号声明类伪指令 .....           | 172        |
| 9.3        | 数据定义类伪指令 .....           | 174        |
| 9.4        | 控制程序流向的伪指令 .....         | 179        |
| 9.5        | 其他伪指令 .....              | 185        |

|                                        |     |
|----------------------------------------|-----|
| 第 10 章 C 语言编程方法 .....                  | 189 |
| 10.1 C 语言的两种编程方法 .....                 | 189 |
| 10.2 创建一个 C 语言工程项目 .....               | 191 |
| 10.3 寄存器编程方法 .....                     | 194 |
| 10.3.1 寄存器定义和寄存器赋值 .....               | 194 |
| 10.3.2 逻辑判断语句的应用 .....                 | 198 |
| 10.3.3 Contex-M3 的位段操作在 C 语言中的应用 ..... | 199 |
| 10.4 工程项目组成 .....                      | 200 |
| 10.4.1 创建和添加一个扩展名为 h 的头文件 .....        | 201 |
| 10.4.2 创建和添加一个中断服务程序文件 .....           | 201 |
| 第 11 章 混合编程方法 .....                    | 204 |
| 11.1 嵌入式汇编语法 .....                     | 204 |
| 11.2 参数传递 .....                        | 206 |
| 第 12 章 通用外部端口 GPIO 和复用外部端口 AFIO .....  | 209 |
| 12.1 通用输入/输出端口概述 .....                 | 209 |
| 12.2 通用 I/O 端口应用 .....                 | 212 |
| 12.3 复用 I/O 端口 .....                   | 218 |
| 12.4 复用功能的重新映射 .....                   | 221 |
| 12.5 寄存器组说明 .....                      | 225 |
| 第 13 章 模拟/数字转换器 ADC .....              | 227 |
| 13.1 ADC 信号源 .....                     | 228 |
| 13.2 A/D 信号源分组 .....                   | 228 |
| 13.2.1 概 述 .....                       | 228 |
| 13.2.2 信号源分组 .....                     | 229 |
| 13.2.3 分组寄存器介绍 .....                   | 231 |
| 13.2.4 分组方法举例 .....                    | 234 |
| 13.3 使用 ADC 的几个概念 .....                | 235 |
| 13.3.1 采样时间及采样时间寄存器 .....              | 235 |
| 13.3.2 数据对齐 .....                      | 237 |

|               |                        |            |
|---------------|------------------------|------------|
| 13.3.3        | 校 准                    | 238        |
| 13.3.4        | ADC 时钟                 | 238        |
| 13.3.5        | 内部触发和外部触发              | 239        |
| 13.3.6        | 中 断                    | 240        |
| 13.3.7        | ADC 复位后状态              | 241        |
| 13.4          | 单次转换和连续转换              | 242        |
| 13.5          | 扫描模式                   | 243        |
| 13.6          | 间断模式                   | 245        |
| 13.7          | ADC 的其他特性              | 247        |
| 13.7.1        | 模拟看门狗                  | 247        |
| 13.7.2        | 双 ADC 模式               | 248        |
| 13.7.3        | 不同型号产品 ADC 的区别         | 252        |
| 13.8          | ADC 寄存器组               | 253        |
| 13.9          | ADC 程序实例               | 260        |
| <b>第 14 章</b> | <b>通用异步同步收发器 USART</b> | <b>264</b> |
| 14.1          | 异步通信概述                 | 264        |
| 14.2          | 波特率设置                  | 266        |
| 14.3          | 数据发送                   | 267        |
| 14.4          | 数据接收                   | 269        |
| 14.5          | 异步串行通信中的中断             | 270        |
| 14.6          | 异步串行通信应用举例             | 271        |
| 14.7          | USART 寄存器介绍            | 273        |
| <b>参考文献</b>   |                        | <b>276</b> |

# 第 1 章

## 概 述

本章简单介绍 STM32F1xx 系列的特性和选型方法,介绍这个系列芯片的分类和区别。

### 1.1 STM32F1xx 系列概述

#### 1.1.1 STM32F1xx 的多个子系列

STM32F1xx 系列高性能、低价格芯片是 ST 公司的产品。这个产品系列又分为多个子系列,它们分别是 STM32F100、STM32F101、STM32F102、STM32F103、STM32F105 和 STM32F107。这些子系列的区别如下:

① STM32F100 子系列:这个系列是最初的产品,与其他系列相比主要是速度低,时钟速度是 24 MHz,其他没有大的区别。

② STM32F101 子系列:这个系列是 STM32F1xx 系列中的“基本型”产品,是一种 16 位的微处理器,也具有 32 位的性能。速度有所提高,时钟达 36 MHz。

③ STM32F102 子系列:这个系列增加了 USB 接口,称为 USB 基本型,可以满足 USB2.0 标准的要求,为实际应用增加了 USB 选择。

④ STM32F103 子系列:这个系列是 STM32F1xx 系列中的“增强型”产品,具有完全的 32 位的处理器性能、丰富的外部接口、强大的数据处理能力,适合于低电压/低功耗的应用场合。

⑤ STM32F105 子系列和 STM32F107 子系列:这个系列增加了 Internet 网络接口,称为“互联型”系列,可以满足微型设备的网络连接需要。

处理器的时钟速度是各子系列的重要区别,如表 1-1 所列。

表 1-1 STM32F1xx 子系列处理器速度的比较

| 型 号       | 处理器速度/MHz | 引脚数              |
|-----------|-----------|------------------|
| STM32F100 | 24        | 48 64 100 144    |
| STM32F101 | 36        | 48 64 100 144 36 |
| STM32F102 | 48        | 48 64            |

## 第 1 章 概 述

续表 1-1

| 型 号       | 处理器速度/MHz | 引脚数              |
|-----------|-----------|------------------|
| STM32F103 | 72        | 48 64 100 144 36 |
| STM32F105 | 72        | 64 100           |
| STM32F107 | 72        | 64 100           |

表 1-2 列举了各子系列的性能区别,其中三相定时器可以用来控制步进电机。所有系列中所包含的功能没有列举在表内。

表 1-2 STM32F1xx 系列外设比较

| 型 号           | 最大存储器/KB |      | USB 2.0 | CAN 2.0B | 以太网 | 三相<br>定时器 |
|---------------|----------|------|---------|----------|-----|-----------|
|               | Flash    | SRAM |         |          |     |           |
| STM32F100     | 512      | 32   |         |          |     | 有         |
| STN32F101     | 512      | 80   |         |          |     |           |
| STM32F102     | 128      | 16   | 有       |          |     |           |
| STM32F103     | 512      | 96   | 有       | 有        |     | 有         |
| STM32F105/107 | 256      | 64   | 有       | 有        | 有   | 有         |

这些子系列除了上述区别外,它们的兼容性主要有:

- 封装相同的芯片引脚兼容。
- 应用程序兼容。
- 开发工具和开发环境相同。
- Flash 写入方法和工具兼容。

因为有以上兼容性,所以封装相同的芯片可以互换。

### 1.1.2 STM32F1xx 系列特点

STM32F1xx 系列包含多个子系列,这些产品不尽相同,各有特色,主要共性和特点如下:

- 采用标准的 ARM Cortex-M3 内核。
- 体积小,采用高密度的方形封装。
- 耗电低;电压低,范围是 2.0~3.6 V。
- 接口丰富,各种常用接口。
- 最高达 72 MHz 的时钟速度。
- 引脚耐压高:I/O 口可以允许 5 V 电压。
- 内嵌 Flash 程序存储器,容量从 32~1 000 KB。
- 内嵌 RAM 数据存储器,容量从 6~96 KB。

- JTAG 调试接口,通用化程度高,开发环境成熟。
- 不需要外部器件,可以独立组成系统,单片化程度高。
- 内嵌 43 个可屏蔽的中断通道(对于小容量芯片)。
- 内部包含 RC 时钟振荡器,可以省略外部振荡器。
- 成本低,可以替代低端单片机。
- 适合裸机环境(不使用操作系统)开发。

综合以上特点,STM32F1xx 系列的产品是一种高性能、低价格的产品,特别适用于中小设备的应用,尤其适合微型设备、仪表和其他电子产品的应用。它可以替代低端的单片机,以提高产品的性能和档次。

STM32F1xx 系列不适于使用操作系统,不能运行大尺寸的软件程序。芯片根据 Flash 的大小分为低密度(小容量 16~32 KB),中密度(中等容量 64~128 KB)和大密度(大容量 256 KB 以上)三个级别,中小容量产品没有外部总线接口,不支持外部存储器。

### 1.1.3 内核简介

STM32F1xx 系列微处理器的内核是一个 ARM 结构,称为 ARM Cortex-M3。这是一个 ARM 的高端产品。众所周知,ARM 是一个 32 位的微处理器结构,这个结构从 ARM1 开始逐步升级直至 ARM11,之后不再沿用以前的叫法。ARM11 的升级产品称为 ARM Cortex,因此 ARM Cortex 是一个高于 ARM11 的结构。

ARM Cortex 结构分为三个系列,分别是 ARM Cortex-A、ARM Cortex-H 和 ARM Cortex-M。STM32F1xx 系列微处理器的内核是 Cortex-M 结构,这个结构是 ARM 公司在 ARM Cortex 等级上设计的一个专用内核结构,主要应用于微型、廉价的小型设备、手提设备等,以满足低端产品的需要。

Cortex-M 结构包含四个系列,它们分别是 Cortex-M0、Cortex-M1、Cortex-M3 和 Cortex-M4,STM32F1xx 系列内核是 Cortex-M3。

Cortex-M3 具有三级流水线,完全执行 Thumb-2 指令集和传统的 Thumb 指令集,已不再兼容 ARM 指令集,不支持浮点运算指令。Cortex-M3 结构不支持 Caches 存储器,Thumb-2 指令比 Thumb 指令更加丰富,其增加了除法指令。

## 1.2 封装

STM32F1xx 作为一个广泛应用的比较成功的微处理器,有多种封装形式,可以满足不同的设计需要,其芯片封装外形图如图 1-1 所示。

其封装主要有 LQFP-48、LQFP-64、LQFP-100、BGA-100、LQFP-144 和 BGA-144 类型,也有 LQFP-36 的小封装。

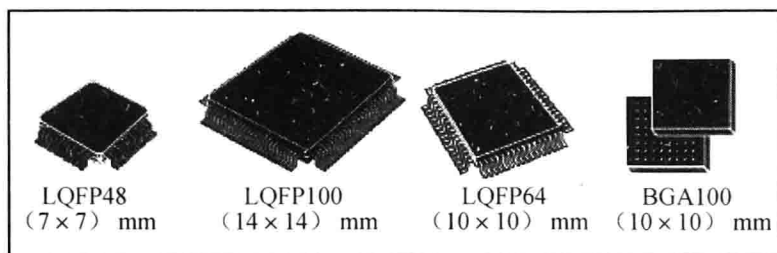


图 1-1 芯片封装外形图

引脚多的封装芯片具有较强的功能,比如 64 个引脚以上的芯片具有多达 5 个 USART 接口,同时引脚多的芯片也内嵌较大的存储器。

### 1.3 接 口

STM32F1xx 系列作为 ARM 的高端产品,和其他微处理器相比,尽管在体积、引脚上做了极大的削减,但大部分功能依然保留,因此接口功能很丰富。常用的接口基本都有,列举如下:

- 通用输入/输出(I/O)接口,最多可达 6 组。
- 12 位精度的 A/D 转换器。
- 标准的 JTAG 调试接口。
- 多个通用定时器和多个高级控制定时器。
- 2 个 I<sup>2</sup>C 接口。
- 多个异步/同步 USART 接口,支持局域网 LIN 和调试模式 IrDA。
- 多个同步串行 SPI 接口。
- 一个 2.0B 标准的 CAN 接口。
- 一个全速的 2.0 标准的 USB 接口。
- 微型卡 SDIO 接口。
- “互联型”型号中内嵌 Internet 网络接口。

并不是每种芯片都包含这些接口,不同的子系列所包含的接口有所不同。增强型芯片包含的接口更多一些,基本型 STM32F101 所包含的接口要少一些,而 STM32F100 就更少一些。

这些接口都具有复用功能,可以作为通用 I/O 口使用,而且大部分都具有重映射用法。重映射用法允许这些接口的默认引脚在和其他使用发生冲突时,使用另一组引脚实现这些功能。因此很多接口有两组引脚可供选用,从而使设计工作更加灵活。图 1-2 所示为 ARM 公司的技术。

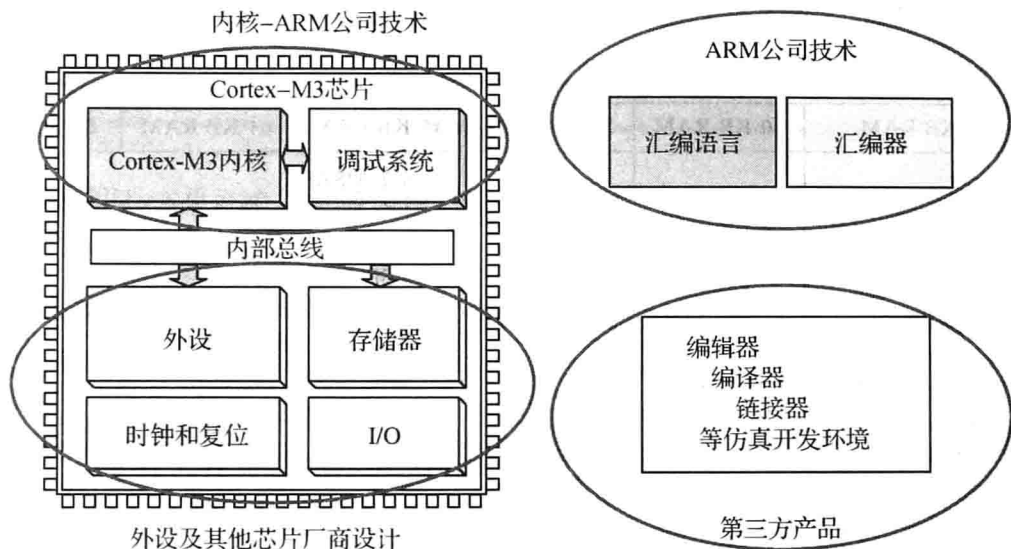


图 1-2 ARM 公司的技术

## 1.4 选型指南

STM32F1xx 系列包含多个子系列,每个子系列 都有多种型号,每种型号都有自己的特点,因此有十分丰富的型号可供选择。和其他高性能的微处理器比较,STM32F1xx 内嵌了程序存储器 Flash,单片化程度比较高,因此十分有利于在小型设备、仪表上的应用。

一般选型规则:

① 根据产品任务要求确定微处理器所需要的接口种类。比如是否需要 SPI 接口,是否需要 CAN 接口或 USB 接口等。根据接口种类确定能够满足要求的芯片种类。

② 根据产品任务确定所需要的通用 I/O 口数量,由此进一步确定能够满足要求的芯片类型。

表 1-3 是 STM32F103 系列产品性能选型表,主要列举性能和引脚以及内部存储器之间的关系。可以通过这个表,根据所要求的性能选定芯片的引脚数和内存容量。

表 1-3 STM32F103 系列产品性能选型表

| 引<br>脚<br>数 | 存储器                                                                 |                                                                                               |           |                                                                                                                                                                |           |           |
|-------------|---------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|-----------|
|             | 32 KB 闪存                                                            | 64 KB 闪存                                                                                      | 128 KB 闪存 | 256 KB 闪存                                                                                                                                                      | 384 KB 闪存 | 512 KB 闪存 |
|             | 10 KB RAM                                                           | 20 KB RAM                                                                                     | 20 KB RAM | 48 KB RAM                                                                                                                                                      | 64 KB RAM | 64 KB RAM |
| 144         |                                                                     |                                                                                               |           | 5 个 USART<br>4 个 16 位定时器、2 个基本定时器<br>3 个 SPI、2 个 I <sup>2</sup> S、2 个 I <sup>2</sup> C<br>USB、CAN、2 个 PWM 定时器<br>3 个 ADC、1 个 DAC、1 个 SDIO<br>FSMC(100 和 144 脚) |           |           |
| 100         |                                                                     | 3 个 USART<br>3 个 16 位定时器<br>2 个 SPI、2 个 I <sup>2</sup> C<br>USB、CAN<br>1 个 PWM 定时器<br>1 个 ADC |           |                                                                                                                                                                |           |           |
| 64          | 2 个 USART<br>2 个 16 位定时器<br>1 个 SPI、1 个 I <sup>2</sup> C<br>USB、CAN |                                                                                               |           |                                                                                                                                                                |           |           |
| 48          | 1 个 PWM 定时器<br>1 个 ADC                                              |                                                                                               |           |                                                                                                                                                                |           |           |
| 36          |                                                                     |                                                                                               |           |                                                                                                                                                                |           |           |

表 1-4 是 STM32F103 系列产品型号和引脚数对照表,可以通过这个表,根据引脚数和内存容量确定所需要的产品型号。

表 1-4 STM32F103 系列产品型号和引脚数对照表

| 类 别    | STM32F103Cx |     |     | STM32F103Rx |     |     | STM32F103Vx    |     |     | STM32F103Zx    |     |     |
|--------|-------------|-----|-----|-------------|-----|-----|----------------|-----|-----|----------------|-----|-----|
| 闪存/KB  | 256         | 384 | 512 | 256         | 384 | 512 | 256            | 384 | 512 | 256            | 384 | 512 |
| RAM/KB | 48          | 64  |     | 48          | 64  |     | 48             | 64  |     | 48             | 64  |     |
| 封 装    | LQFP48      |     |     | LQFP64      |     |     | LQFP100,BGA100 |     |     | LQFP144,BGA144 |     |     |

图 1-3 是 STM32F103 系列产品订货代码表达法,要确定自己所选定的型号,准确表达型号代码。图中列举 13 位代码,后面的代码一般可以不考虑,主要表达芯片的装箱形式和每个最小包装的数量。

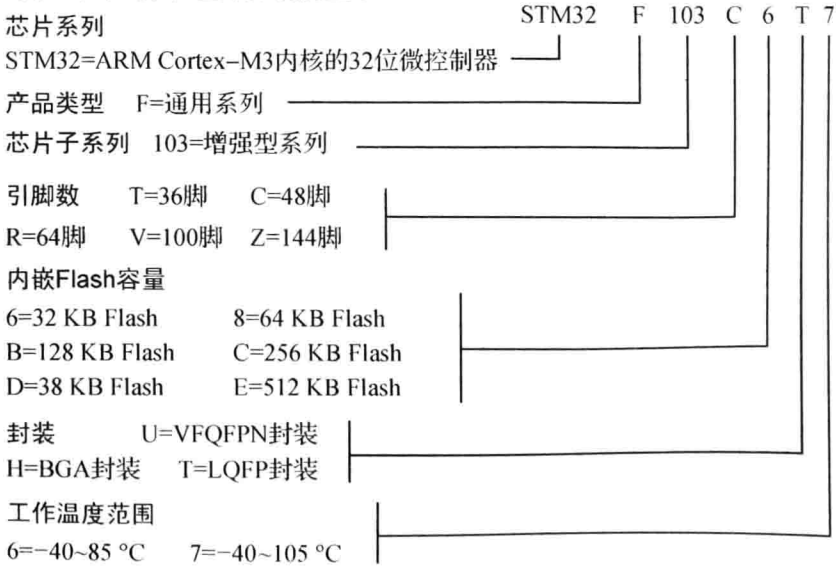


图 1-3 STM32F103 系列产品订货代码表达法